Proceedings of the 53rd Hawaii International Conference on System Sciences | 2020

Towards an Integrative Approach for Automated Literature Reviews Using Machine Learning

Christoph Tauchert
TU Darmstadt
tauchert@is.tu-darmstadt.de

Marco Bender
TU München
bender.marco@gmail.com

Neda Mesbah
TU Darmstadt
mesbah@is.tu-darmstadt.de

Peter Buxmann
TU Darmstadt
buxmann@is.tu-darmstadt.de

Abstract

Due to a huge amount of scientific publications which are mostly stored as unstructured data, complexity and workload of the fundamental process of literature reviews increase constantly. Based on previous literature, we develop an artifact that partially automates the literature review process from collecting articles up to their evaluation. This artifact uses a custom crawler, the word2vec algorithm, LDA topic modeling, rapid automatic keyword extraction, and agglomerative hierarchical clustering to enable the automatic acquisition, processing, and clustering of relevant literature and subsequent graphical presentation of the results using illustrations such as dendrograms. Moreover, the artifact provides information on which topics each cluster addresses and which keywords they contain. We evaluate our artifact based on an exemplary set of 308 publications. Our findings indicate that the developed artifact delivers better results than known previous approaches and can be a helpful tool to support researchers in conducting literature reviews.

1. Introduction

Due to the advancing digitization, more and more data is being generated in a wide variety of areas, including science [9]. For example, in mid-2019 over 560,000 documents are found in all EbscoHost databases for the keyword search "artificial intelligence" (AI). The number of scientific publications is increasing immensely. Although these papers are mostly accessible, the information is prevalently unstructured (i.e., available as PDF file) [32]. A fundamental task of researchers is to discover and understand the existing literature through a literature review in order to establish the context and conduct new and further research [14]. For this purpose, it is essential that all existing literature relating to a research topic is reviewed. However, this task is hardly feasible with the constantly increasing number of papers and their evaluation is practically difficult. To cope with the huge amount of publications, researchers might be supported by an IT artifact for structured literature reviews, which collects available documents and provides first insights into the existing literature.

Recent developments in technology, especially in machine learning, enabled (partially) automated literature reviews to become technically feasible. AI is a sub-field of computer science containing techniques such as machine learning, deep learning, and natural language processing to enable intelligent machines [17, 29]. AI is efficient and scalable [11] and provides capabilities to enable a machine to process more information and gain deeper insights than any human being can because of their cognitive constraints [39]. In the past, several attempts to use data mining to solve specialized problems similar to automated literature reviews (e.g., medical case analysis [23]) have been made. However, there is still no well-established method of how this new technology can be used to perform a (partially) automated literature review. We, therefore, try to address the research question: How can an IT-artifact be designed to support researchers in conducting structured literature reviews?

To our best knowledge, only Dann et al. [14] developed an artifact that uses the word2vec algorithm and keyword extraction to automate the literature review process based on full-text papers. Nevertheless, their approach has weaknesses. For example, each paper still has to be downloaded manually, which is a challenge with these large quantities of papers. Additionally, identifying the theme of a cluster is not easy and still involves a lot of work.

The aim of our research is to extend their approach

URI: https://hdl.handle.net/10125/63834
978-0-9981331-3-3

so that the whole process from collecting the data, processing and evaluating the clusters becomes simpler and more reliable.

To achieve this goal, we first sum up the related literature, where we focus in particular on the approach of Dann et al. [14]. Then we describe the design science method on which we have based and further developed the artifact. Afterward, we present and evaluate our artifact. Finally, we summarize our results.

2. Related Research

Literature reviews play a crucial role in research and science since the creation of new knowledge is often based on the interpretation, combination, and questioning of already existing knowledge [37]. However, conducting a literature review is very time consuming and cumbersome due to the many manual activities, such as searching and downloading, documentation of the process, text screening, etc. Nonetheless, knowing and understanding the results and findings of existing literature is crucial to contribute to research and helps to avoid investigating what has already been investigated [37].

One of the most renowned and widely-used process models in IS research for conducting a (manual) literature review is the framework of vom Brocke et al. [10], which is depicted in Figure 1. This framework is often used in conjunction with the concept matrix suggested by Webster and Watson [42]. This matrix helps to understand and link the various concepts used in the processed publications.

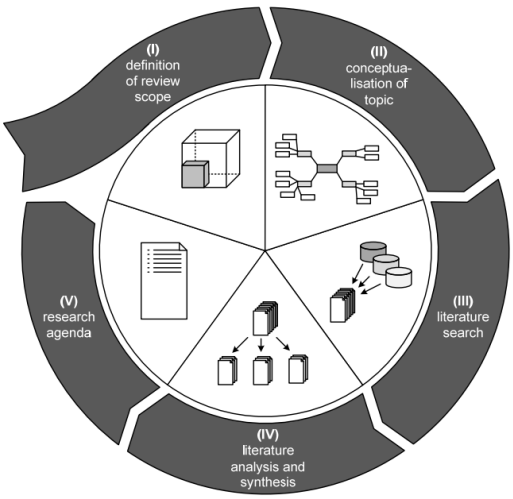

Figure 1. Framework for literature reviews according to vom Brocke et al. [10]

Regarding the usage of algorithms to analyze scientific documents, Dann et al. identified three categories: (1) citation-based approaches which only consider the links between documents by analyzing the references, (2) text-based approaches which analyze the actual textual context of the documents and (3) hybrid approaches which combine the two former mentioned approaches [14].

Since we focus on a content-based analysis to extract knowledge from existing literature, we are only considering manuscripts that deal with text-based approaches. Furthermore, text-based approaches are considered superior to citation-based ones for document categorization [3]. The used approaches differ in three aspects: (1) text sections (i.e., abstract, keywords, full text), (2) objective (e.g., classification, recommendation, content extraction, clustering), and (3) used techniques (e.g., bag-of-words, vectorization, Bayesian classifier, topic models, keyword extraction) [1, 14, 20, 41].

While we were inspired by and based our artifact on Dann et al.'s [14] presented process, we suggest an extension of their model to improve information extraction and automation. By implementing a crawler to download all documents related to a search term automatically, a very effortful but rather trivial task is automated. Furthermore, by implementing a topic model using Latent Dirichlet Allocation (LDA) we add another analytical layer to gain more insights into the gathered literature. The extracted topics can especially be helpful to identify common concepts of the scientific publications supporting the derivation of Webster and Watson's concept matrix.

3. Design Science Research

The proposed solution is an IT artifact in accordance with Hevner et al. [22]. Due to the fact that the proposed solution solves a problem that is primarily targeted and based on the "*science business*", it can be categorized as an idiographic design science artifact since it is an "ideal artifact for a specific problem" [5]. Identifying and analyzing a manifold of databases, searching relevant literature and analyzing content is, on the one hand, a very time-consuming task for scientists but a partly structured task with some repeating actions on the other hand which makes it suitable for automation – not just in the science context (e.g., [21]). Developing a solution, which partially supports or entirely replaces this part of the research work is therefore highly relevant for the scientific community. The growing number of publications in journals or conference proceedings as well as other potentially less scientific sources poses a great challenge to researchers since a comprehensive and extensive overview of a certain topic becomes harder to attain [3, 9, 23]. Data-mining techniques have the

potential to overcome these challenges and especially text-mining can be applied to the unstructured data that scientific publications usually contain [3, 16]. Current research and knowledge discovery processes generally have a very low degree of automation and are vastly done by humans instead of algorithms since these tasks are usually less structured compared to manufacturing processes, for example.

Therefore, the support of a software-based, semi-automated knowledge extraction tool has several (practical) benefits for the researcher and can overcome the described restrictions:

- *time*: software-based solutions can gather and analyze publications faster than humans. Furthermore, the process can easily be parallelized and is therefore scalable.
- *structure*: the research process is always executed identically. Also, every publication is analyzed in the same way and results are processed alike. This guarantees repeatability and reduces subjectivity and personal bias in the evaluation process [25].
- *cross-disciplinarily*: although the way research is conducted varies across several science areas, the preparation of a research project is in most cases at least similar. This also means that the automated process can be transferred across disciplines and is ideally not restricted to a specific, singular discipline, like Information Systems (IS) research. The positive effects of this standardization and automation are increased comparability and eventually, improvements in the generation of insights and explanations can be achieved in a standardized way [28].

The overall implementation is similar to the process described by Dann et al. [14]. An extension our solution provides is the addition of the actual information retrieval process. Relevant publications are identified by using the search interfaces of online databases and full texts as well as bibliographic information are downloaded automatically. This not only speeds up the whole research preparation phase, but it also enables further filtering after the contents are locally available and document selection/filtering is no longer restricted to the capabilities of the database and does not require manual inspection and evaluation of whether a document fits the required search criteria or not. The Python programming language was chosen to implement the software artifacts since it provides platform independence and many readily available packages that are common in the data science process. The following steps of data preparation and processing are implemented analogously to established text-mining processes: conversion of full-text PDF files to machine-readable text files, vectorization of text files, clustering, and keyword extraction (e.g., Dann et al. [14]). At the end of the process, the results are visualized and presented to the user.

The artifact was developed in an iterative manner. Initial requirements were successively extended and the solution was implemented and tested according to the additional requirements. It, therefore, is designed as a search process in which an initial solution was continuously enhanced and refined to reflect the process of preparation steps to knowledge extraction better step by step [22].

Since there is little guidance in the IS literature on how to evaluate design-science research [34], our approach for the evaluation of the artifact is achieved in a three-stage process. First, the artifact was used in a specific context, i.e. the actual functionality of the data acquisition, filtering, pre-processing and clustering was evaluated. Second, the results of the first steps were evaluated by humans who determined whether the proposed clusters are correct and useful. Lastly, the clusters themselves were discussed by a group of four IS researchers [22].

4. ALR Approach

Our artifact contains of the following steps: (1) downloading documents and making documents machine-readable, (2) preprocessing downloaded full-text documents, (3) vectorising documents, (4) extracting keywords, (5) identifying topics, (6) clustering documents and (7) visualization (see Figure 2).

As many data science projects are written in Python, our artifact likewise is also largely implemented in Python. The data acquisition is separated into several classes since it is a more complex task. The process was then orchestrated in Jupyter notebooks, which are a convenient way to combine code and documentation. At this point, efficiency and performance were not the main goals, instead, the focus lied on building an easily readable and reusable code base with an understandable interface. Jupyter notebooks also allow for rapid code changes and integrating visualization.

4.1. Collecting and Generating Machine-Readable Documents

The first element of our artifact has the purpose of downloading the documents that will be analyzed in the subsequent process steps. Therefore, we wrote a web crawler for two established scientific databases (i.e., EbscoHost and ScienceDirect), that downloads all

available publications that match the user-defined *search terms* and *date range*. While ScienceDirect offers an API to search and download plain text documents, we used the Python library *Selenium* to download documents from EbscoHost. This library allows to "remote control" a web browser to navigate the EbscoHost website and extract the required information without an API.

the process can be conducted: the preprocessing of the textual data.

Depending on the search criteria (i.e., search terms and date range) specified in step 1, the automated download of documents may lead to a large number of documents, which the user might want to restrict in retrospect. For that reason, we added the possibility to reduce the set of documents using *filter terms*. All

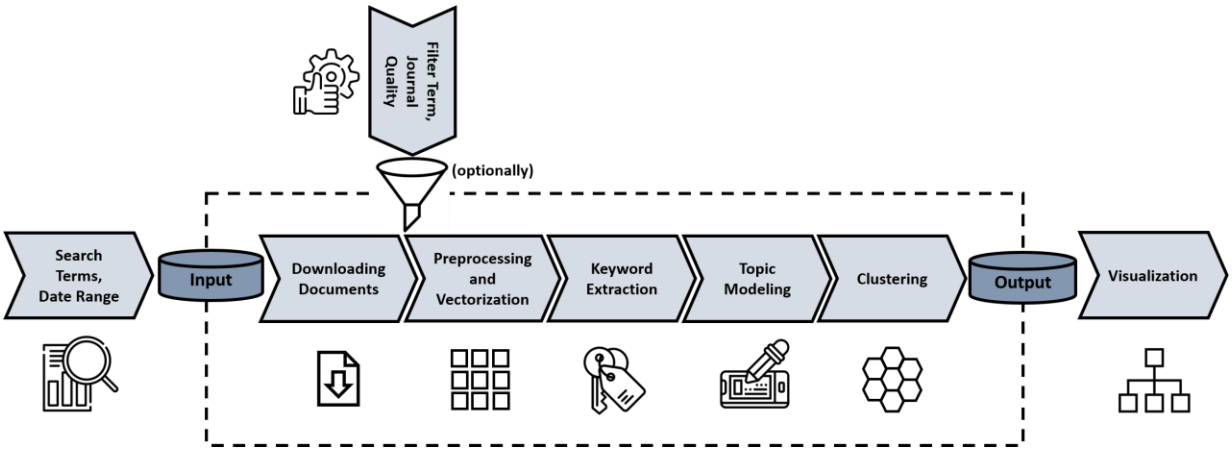

Figure 1. Automatic literature review process

In contrast to the ScienceDirect API, documents that were downloaded through EbscoHost were only available as PDF files and were therefore not directly machine-readable. Older manuscripts are often embedded images of scanned manuscripts, which cannot be directly processed by a text-mining algorithm. Similarly, newer files usually contain the plain text version of the publication but due to the proprietary binary format, it is difficult to extract the text using open source libraries. To overcome these issues, we used the open-source optical character recognition (OCR) library *Tesseract* that can convert the PDF files into plain text. Tesseract is a popular and widespread software for OCR and is currently developed by Google, whose engineers utilize it themselves for text recognition on mobile devices, for example. The output of Tesseract is a plain text file comparable to the files that were retrieved via the ScienceDirect API.

The result of this step is a collection of plain text documents of all scientific documents available through EbscoHost and ScienceDirect. Furthermore, the program uses a database to store metadata about the retrieved documents such as title, authors, year, journal, etc.

4.2. Text Preprocessing

With all relevant documents available in a machine-readable plain text format, the next subset of

documents, which do not contain the *filter terms*, are then excluded from further processing. Therefore, filter terms can also be used to change the scope of the analysis iteratively based on the insights gained through previous analysis, e.g., extracted keywords, topics or frequent words. The effectiveness of reducing the document set by using filter terms depends largely on the specificity of the filter terms and the heterogeneity of the documents. Additionally, we implemented the option to consider only documents from journals that have a Q1 score in the Scimago Journal Ranking. This enables the user to limit the analysis to results from journals with a certain quality. Both filtering features are optional and the users can decide whether they want to exclude documents based on filter terms and/or journal ranking.

The preparation of documents for analysis usually contains a text-cleaning step. Hereby, punctuation and stop words (e.g., *for*, *and*, *of*, etc.) as well as user-defined words are removed from the text corpora. In our case, this list contained words such as journal names, placeholders for figures, etc. Furthermore, words are normalized to their word stem (e.g., *fisher*, *fishing*, *fishy* are reduced to *fish*). The result of this step is cleaned and stemmed textual data.

4.3. Vectorization of Documents

To apply a hierarchical clustering approach to the corpus of the collected documents, their text needs to

be represented by vectors of a fixed length. One of the most applied models to transform the representation of documents into vectors is the bag-of-words model. Despite its simplicity and efficiency, it often achieves high accuracy. Texts are represented as an unsorted collection of the contained words. The model then assigns weights to the words, which represent the frequency in the document and in the collection of documents. Documents with similar word frequencies can be considered as having similar contents [18]. The simplicity of the model also yields some major drawbacks. The model does not consider the order of words, which leads to the problem that different semantics of sentences with different order cannot be distinguished. Furthermore, ambiguity and synonyms cannot be considered. Ambiguity means that words can have different meanings depending on the surrounding context. Synonyms are words that are different but do have the same meaning.

Due to the aforementioned drawbacks of the bag-of-words model, we generate the vector representation by using the paragraph-vector model which is based on the word2vec model [26, 31]. These models take the context of words into account (i.e., the paragraph) and therefore partly solve the issues of the bag-of-words model. The paragraph-vector algorithm learns continuously distributed vector representations of texts of any length by using artificial neural networks to learn a word vector for any word in the document collection [2, 26]. Each document can, therefore, be represented as a structured concatenation of word vectors.

The calculated word vector representation for the documents allows to compare similarities of documents by using common distance measures. We calculate a document **X** document distance matrix, which can then be used for hierarchical clustering.

4.4. Keyword Extraction

Keywords are often used to tag documents for the purpose of information retrieval [35]. There are many available approaches to automatically generate keywords, which can be categorized in statistical, supervised, semi-supervised and unsupervised approaches [38]. In our context, only statistical and unsupervised methods can be used, since the downloaded documents are not labeled.

An algorithm for keyword extraction that is often used in practice is RAKE (Rapid Automatic Keyword Extraction) [35].

RAKE assumes that keywords frequently consist of multiple words but rarely contain standard punctuation or stop words (i.e., *and*, *the*, and *of)*, or other words with minimal lexical meaning. Therefore, the algorithm uses these stop words and phrase delimiters to create a list of candidate keywords by partitioning the text at these positions. Afterward, the graph of co-occurrences is computed and word scores are calculated. The word score is the ratio of word degree and word frequency. The word degree is the sum of co-occurrences and favors words that occur often and in longer candidate keywords. The word frequency is the pure number of occurrences of a word in the candidate list. Due to the partition of the text using stop words and phrase delimiters, candidate keywords cannot contain any stop words, such as in *illusion of control*. To tackle this problem the algorithm then searches for pairs of keywords and creates a combination of two keyword candidates if they adjoin one another at least twice in the same document. Afterward, the *N* top scoring candidates are selected. A proposed number of keywords is one-third the number of words in the graph of co-occurrences [30].

4.5. Topic Identification

To gain more insights into the clustered documents, we added another analytical layer to the process. By implementing a probabilistic topic modeling approach, we can get an overview of predominant topics within the clusters. In our case, we use LDA as topic model. LDA assumes that documents cover multiple topics, which can be seen as a distribution over a defined vocabulary [8]. Furthermore, it is assumed that topics existed even before the documents were written. LDA tries to invert the imaginary random process and therefore guesses which hidden topic structure has probably generated the observed document collection [7, 8].

To use LDA, we have to specify the number of topics that should be identified. There are several evaluation metrics to assess the appropriateness of a topic model [4, 13, 15, 19]. A typical approach is to calculate multiple metrics and determine the number of topics by aggregating the provided information.

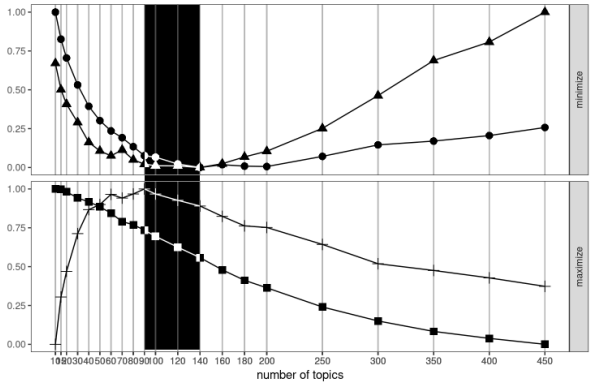

Figure 3: Selecting the number of topics [33]

Figure 3 exemplary shows the result of the calculation of the four metrics for 10 to 450 topics for the Associated Press data set. While the metric by Deveaud et al. [15] is not informative in this case, the other three metrics reach their minima or maxima in the area of 90 to 140 topics.

The result of this step is a topic model, a list of identified topics and their related words as well as the information which topics are present in each document.

4.6. Clustering of Documents

Clustering is a very popular approach when it comes to text mining and its goal is to form groups of similar documents by detecting hidden patterns within them. All documents contained in one cluster should be similar to the other documents in the same cluster but different from documents contained in every other cluster [6].

One popular clustering algorithms is *k-means clustering*, which is widely used in data mining. This algorithm creates *k* clusters, with *k* being a number that has to be specified beforehand. It then maximizes the sum of squared deviations between documents in different clusters [24].

While there are multiple methods to estimate the optimal number of clusters, such as the elbow criteria [40] and silhouette score [36], we did not get useful results in our context. The elbow criteria usually suggested forming two clusters while the silhouette score preferred as many clusters as possible.

Therefore, we decided to use a more flexible approach without the need to specify the number of clusters beforehand: agglomerative hierarchical clustering. Hereby, the algorithm calculates a tree-like hierarchy, which can easily be visualized using a dendrogram and enables an explorative data analysis with varying granularity. Agglomerative clustering initially creates a cluster for every object and recursively joins those clusters until a cutoff-value for the distance between the clusters is reached [27].

4.6. Visualization of Analysis

Since the set of documents usually is rather large, depending on the chosen search and filter terms, we chose to adapt the representation of the results. While there is also information provided on the entire data set, the more detailed information such as keywords and topics are only provided within the generated clusters. This approach counter-acts information overload [12].

The generally provided information consists of an overview of (1) the distribution of documents across clusters, (2) which journals are represented, (3) the distribution of publishing dates and (4) the identified topics.

The information per cluster consists of: (1) number of documents in cluster, (2) number of identified LDA topics, (3) range of publishing dates, (4) distribution of LDA topics, (5), extracted keywords, (6) most common words and two-word phrases, (7) represented journals in the cluster, (8) dendrogram with titles, (9) dendrogram with LDA topics, (10) dendrogram with keywords, (11) dendrogram with authors.

5. Evaluation

For the initial evaluation of our approach, we decided to use a rather small set of documents. This allows us to assess the quality of generated keywords, topics and clusters manually and properly. Although the document crawler was working as intended, we used an existing set of 308 documents on the *application and usage of sensor data in industrial manufacturing* due to the occurrence of licensing issues and resulting time restrictions.

Figure 4 shows the distribution of documents across the 17 generated clusters with cluster sizes ranging from nine to 29. Due to the used approach of agglomerative hierarchical clustering, the created partition is not the “one ground truth” but one of many possible partitions. Other partitions might be more specific or more general.

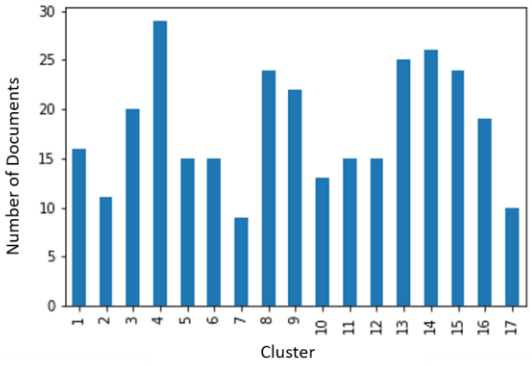

Figure 4. Distribution of documents across clusters

To get an overview of the clusters and therefore the similarity of the documents, a dendrogram depicts the distances between the documents and clusters (see Figure 7 in Appendix). We added titles for each cluster, which we derived by inspecting the algorithmically extracted information (i.e., keywords, the most common words and topics). For the LDA topic model, we decided to use 33 topics since the evaluation metrics mentioned in section 4.5 suggested that 25 to 40 topics would be appropriate. Exemplarily

we show three of the found topics and the words they consist of:

1) Real-time RFID technology: “real_time”, “rfid”, “shop_floor”, “material”, “production” “task”, “technology”, “location”, “product” “operator”
2) Intelligent Grinding (with Industry 4.0): “grinding”, “industryfourzero”, “production”, “level”, “wheel”, “rule”, “expert”, “intelligent”, “grinding_wheel”, “rowe”
3) Real-time Fault Detection and Simulation in Assembly: “real_time”, “error”, “event”, “simulation”, “assembly”, “station”, “degradation”, “line”, “exception”, “service”

The remainder of this section describes the information that is visualized (not computed) for every isolated cluster to prevent information overload. Due to space restrictions, we are describing and evaluating the data only for the first cluster *Process Monitoring and Error Diagnostic for Assembly Stations.*

Figure 5 shows the most prevalent topics for the first cluster. The higher the percentage share of the top topics, the more homogenous the cluster is. When all topics have a relatively low percentage share, the cluster is more diverse. We can see that the first cluster is about error detection at assembly stations, tonnage signals and detecting errors on wafers.

Cluster 1: Average LDA Topic Distribution
29.1% Errors at assembly stations
28.7% Error diagnostic with tonnage signals
25.2% Error diagnostic and recognition at wafers
7.1% Algorithms for error and quality classification
3.1% Detecting surface errors with machine vision

Figure 5. LDA topics of cluster 1

This information can be enriched by also considering the automatically extracted keywords. The keywords are partitioned by the number of words and sorted in declining order of their RAKE-score (see Figure 6). Therefore, important keywords are listed first. The number of documents within the cluster that contain this keyword is shown in brackets.

Using the extracted keywords, it can also be seen that this cluster is about monitoring assembly processes mainly in the context of car bodies. Apparently, in many documents principal component analysis is used. The placement of sensors and pattern recognition seem to be of importance.

By looking at the dendrogram (see Figure 8 in Appendix), we can easily understand that the first three documents deal with process monitoring and diagnosis for stamping or forging while the two following documents tackle the problem of thermal errors at machine tools. The analysis of the dendrogram of keywords (not included due to space limitations) shows that these documents were matched because the same approach (i.e., principal component analysis) was used.

Cluster 1: Keyword (number of documents containing the keyword)
state space model(6)
manufacturing process(10), process monitoring(6), principal component(10), assembly process(11), assembly station(8), fault diagnosis(11), fixture layout(9), dimensional quality(7), sensor location(8), locating scheme(4), fixture fault(9), autobody assembly(7), variation pattern(7), pattern recognition(9), engineering knowledge(6), standard deviation(7), sample size(9), dimensional variation(9), measurement data(8), proposed method(12), proposed methodology(5), covariance matrix(8), final product(7), measurement point(7), locating pin(5), degree freedom(5), assumed independent(4)
table(16), distance(7), monitoring(10), line(16), time(13), contribution(9), coordinate(9), application(12), sensor(14), quality(13), based(16), change(14), process(15), limit(13), set(16), component(14), variable(15), design(14), assembly(11), control(13), analysis(14), developed(13), diagnosis(14), position(12), source(10), station(12), performance(12), structure(11), model(15)

Figure 6. Extracted keywords of cluster 1

The documents within the red frame are more homogenous and deal with error diagnosis at assembly stations in manufacturing processes. The topic dendrogram of the cluster (see Figure 9 in Appendix) reveals an anomaly within the cluster since the most important topic of the sixth paper does not appear in the other documents of the cluster.

Summarizing the evaluation, we conclude that the proposed artifact leads to useful results for clustering the selected publications in the context of industrial manufacturing and therefore supported the subsequent analysis and synthesis of the literature effectively. The topics contained within the clusters were described well by the extracted keywords. The extension of introducing an additional topic model has proven useful in further understanding of the extracted topics and differentiation of inter-cluster homogeneity and heterogeneity of topics. The topic model also helped by splitting single clusters into multiple sub-clusters. The cognitive load that is put upon the researcher if clustering is done manually was (subjectively) reduced significantly by using this approach compared to a solely manual literature review of the same number of publications. Manual structuring, synthesizing and describing of these manuscripts would also have required significantly more time. Nonetheless, the task of creating a literature review does not become trivial by just introducing a (partially) automated solution.

The researcher is still required to understand and process the literature and to extract relevant knowledge. Especially a close inspection of the decision rules for assigning a publication to a topic is necessary since they might not always reflect the researcher's own expectations. For example, a cluster that is determined by the association of authors could be less suited than a cluster that is chosen because of the contextual proximity of the publication. These supposed 'misclassifications' (from the researcher's subjective point of view) can always happen. Therefore, manual evaluation of the clusters and associated decision rules is always required. This manual analysis of clusters additionally enables the researcher to further improve the results by identifying related clusters that can be aggregated or find big clusters that can be split into multiple sub-clusters, or split them if the topics included in the overarching cluster are spread too much vice versa. Since the automatic generation of clusters is based on statistical analysis, the decision criteria might differ from a human interpretation since humans tend to interpret the meaning of topics and they do not solely rely on statistics and logical reasoning when structuring content and when assigning items to that structure. In summary, the artifact provides a supporting mechanism to speed up and standardize the process of literature reviews and increases automation of an otherwise entirely manual process to ultimately improve the quality and reproducibility of this important aspect of research.

6. Conclusion

In summary, we have developed an artifact based on the word2vec algorithm, LDA topic modeling, rapid automatic keyword extraction, and agglomerative hierarchical clustering. This artifact is a first step towards simplifying the task of literature reviews within scientific research.

For this purpose, the publications are first collected by a crawler and vectorized afterward. Following this, keywords are extracted and the LDA method is used to identify topics. Finally, the word vectors are used to form clusters. These results are presented graphically, for example in the form of dendrograms.

To evaluate the artifact, we used an exemplary set of 308 scientific publications. As the evaluation showed, our extension is particularly suitable for capturing the topic of clusters without looking directly into each paper in detail. However, even in this case, there are cluster topics that are not obvious at first glance. Looking at the combination of extracted keywords and topics can help to understand the reason for the clustering. This type of clustering also opens up new perspectives on topics that might be clustered due to other aspects, as might be possible at first appearance.

As every study, also the present study and its results are to be seen and interpreted in consideration of certain limitations. The rather small evaluation data set of just 308 full-text papers which were manually checked if the proposed clustering of our model reflects the expectation of IS researchers, can only serve as a starting point for future research. Another limitation results from the conversion in plain text documents since all information stored in images and figures is not considered by the artifact. Furthermore, a more rigorous evaluation of the artifact's utility for researchers during the creation of literature reviews in different contexts should be subject to future research. In addition, it would be interesting to conduct a comparative performance analysis of different topic modeling approaches such as LDA, LSA, pLSA, etc. and to evaluate possible improvements that can be achieved by optimizing the implemented algorithms.

Finally, this work provides an insight into how the knowledge available in unstructured text data can be efficiently organized and used. This approach might support researchers in conducting comprehensive literature reviews through machine learning.

6. Acknowledgments

This research project was funded by the Hessian state ministry "Hessisches Ministerium des Innern und für Sport" in Germany.

7. References

[1] Afonso, A.R., and C.G. Duque, "Automated Text Clustering of Newspaper and Scientific Texts in Brazilian Portuguese: Analysis and Comparison of Methods", *Journal of Information Systems and Technology Management 11*(2), 2014, pp. 415–436.

[2] Ai, Q., L. Yang, J. Guo, and W.B. Croft, "Analysis of the Paragraph Vector Model for Information Retrieval", *Proceedings of the 2016 ACM International Conference on the Theory of Information Retrieval*, 2016, 133–142.

[3] Aljaber, B., N. Stokes, J. Bailey, and J. Pei, "Document clustering of scientific texts using citation contexts", *Information Retrieval 13*(2), 2010, pp. 101–131.

[4] Arun, R., V. Suresh, C.E. Veni Madhavan, and M.N. Narasimha Murthy, "On Finding the Natural Number of Topics with Latent Dirichlet Allocation: Some Observations", Springer, Berlin, Heidelberg (2010), 391–402.

[5] Baskerville, R.L., M. Kaul, and V.C. Storey, "Genres of Inquiry in Design-Science Research: Justification and Evaluation of Knowledge Production", *MIS Quarterly 39*(3),

2015, pp. 541–564.

[6] Berkhin, P., "A Survey of Clustering Data Mining Techniques", In *Grouping Multidimensional Data*. Springer-Verlag, Berlin/Heidelberg, 2006, 25–71.

[7] Blei, D., L. Carin, and D. Dunson, "Probabilistic Topic Models: A focus on graphical model design and applications to document and image analysis.", *IEEE signal processing magazine 27*(6), 2010, pp. 55–65.

[8] Blei, D.M., A.Y. Ng, and M.I. Jordan, "Latent Dirichlet Allocation", *Journal of Machine Learning Research 3*(Jan), 2003, pp. 993–1022.

[9] Bornmann, L., and R. Mutz, "Growth rates of modern science: A bibliometric analysis based on the number of publications and cited references", *Journal of the Association for Information Science and Technology 66*(11), 2015, pp. 2215–2222.

[10] Vom Brocke, J., A. Simons, B. Niehaves, et al., "Reconstructing the Giant: On the Importance of Rigour in Documenting the Literature Search Process", *ECIS 2009 Proceedings*, (2009), Paper 161.

[11] Brundage, M., S. Avin, J. Clark, et al., "The Malicious Use of Artificial Intelligence: Forecasting, Prevention, and Mitigation", 2018.

[12] Burkhard, R.A., "Learning from architects:the difference between knowledge visualization and information visualization", *Proceedings. Eighth International Conference on Information Visualisation, 2004. IV 2004.*, IEEE, 519–524.

[13] Cao, J., T. Xia, J. Li, Y. Zhang, and S. Tang, "A density-based method for adaptive LDA model selection", *Neurocomputing 72*(7–9), 2009, pp. 1775–1781.

[14] Dann, D., M. Hauser, and J. Hanke, "Reconstructing the Giant: Automating the Categorization of Scientific Articles with Deep Learning Techniques", *Proceedings der 13. Internationalen Tagung Wirtschaftsinformatik*, (2017), 1538–1549.

[15] Deveaud, R., E. SanJuan, and P. Bellot, "Accurate and effective latent concept modeling for ad hoc information retrieval", *Document numérique 17*(1), 2014, pp. 61–84.

[16] Dumouchel, B., and J. Demaine, "Knowledge Discovery in the Digital Library: access tools for mining science", *Information Services & Use 26*, 2006, pp. 29–44.

[17] Elliot, B., and W. Andrews, *A Framework for Applying AI in the Enterprise*, 2017.

[18] George, S., and S. Joseph, "Text Classification by Augmenting Bag of Words (BOW) Representation with Co-occurrence Feature", *1*, 2014, pp. 34–38.

[19] Griffiths, T.L., and M. Steyvers, "Finding scientific topics.", *Proceedings of the National Academy of Sciences of the United States of America 101 Suppl*(suppl 1), 2004, pp. 5228–35.

[20] Gulo, C.A.S.J., T.R.P.M. Rúbio, S. Tabassum, and S.G.D. Prado, "Mining Scientific Articles Powered by Machine Learning Techniques", *2015 Imperial College Computing Student Workshop (ICCSW 2015) 49*, 2015, pp. 21–28.

[21] Günther, E., and T. Quandt, "Word Counts and Topic Models", *Digital Journalism 4*(1), 2015, pp. 75–88.

[22] Hevner, A.R., S.T. March, J. Park, and S. Ram, "Design Science in Information Systems Research", *MIS Quarterly 28*(1), 2004, pp. 75–105.

[23] Huang, W., Y. Nakamori, S. Wang, and T. Ma, "Mining scientific literature to predict new relationships", *Intelligent Data Analysis 9*, 2005, pp. 219–234.

[24] Kriegel, H.-P., E. Schubert, and A. Zimek, "The (black) art of runtime evaluation: Are we comparing algorithms or implementations?", *Knowledge and Information Systems 52*(2), 2017, pp. 341–378.

[25] Lacity, M.C., and M.A. Janson, "Understanding Qualitative Data: A Framework of Text Analysis Methods", *Journal of Management Information Systems 11*(2), 1994, pp. 137–155.

[26] Le, Q., and T. Mikolov, "Distributed Representations of Sentences and Documents", *Proceedings of the 31st International Conference on Machine Learning 32*, 2014, 1188–1196.

[27] Madhulatha, T.S., "AN OVERVIEW ON CLUSTERING METHODS", *IOSR Journal of Engineering 2*(4), 2012, pp. 719–725.

[28] Martens, D., and F. Provost, "Explaining Data-Driven Document Classification", *MIS Quarterly 38*(1), 2014, pp. 73–99.

[29] McCarthy, J., "WHAT IS ARTIFICIAL INTELLIGENCE?", 2007.

[30] Mihalcea, R., and P. Tarau, *TextRank: Bringing Order into Texts*,

[31] Mikolov, T., K. Chen, G. Corrado, and J. Dean, *Efficient Estimation of Word Representations in Vector Space*,

[32] Nair, R., and A. Narayanan, *Data and Technology Perspective Benefitting from Big Data Leveraging Unstructured Data Capabilities for Competitive Advantage*, 2012.

[33] Nikita, M., "Select number of topics for LDA model", 2019. https://cran.r-project.org/web/packages/ldatuning/vignettes/topics.html

[34] Pries-Heje, J., R. Baskerville, and J. Venable, "Strategies for design science research evaluation", *16th European Conference on Information Systems, ECIS 2008*, (2008).

[35] Rose, S., D. Engel, N. Cramer, and W. Cowley, "Automatic Keyword Extraction from Individual Documents", In *Text Mining*. John Wiley & Sons, Ltd, Chichester, UK, 2010, 1–20.

[36] Rousseeuw, P.J., "Silhouettes: A graphical aid to the interpretation and validation of cluster analysis", *Journal of Computational and Applied Mathematics 20*, 1987, pp. 53–65.

[37] Schryen, G., G. Wagner, and A. Benlian, "Theory of Knowledge for Literature Reviews : An Epistemological Model , Taxonomy and Empirical Analysis of IS Literature",

Thirty Sixth International Conference on Information Systems, Fort Worth, 2015, pp. 1–22.

[38] Siddiqi, S., and A. Sharan, *Keyword and Keyphrase Extraction Techniques: A Literature Review*, 2015.

[39] Simon, H.A., "Theories of Bounded Rationality", *Decision and organization 1*(1), 1972, pp. 161–176.

[40] Thorndike, R.L., "Who belongs in the family?", *Psychometrika 18*(4), 1953, pp. 267–276.

[41] Wang, C., and D.M. Blei, "Collaborative topic modeling for recommending scientific articles", *Proceedings of the 17th ACM SIGKDD international conference on Knowledge discovery and data mining - KDD '11*, ACM Press (2011), 448.

[42] Webster, J., and R.T. Watson, "Analyzing the Past to Prepare for the Future: Writing a Literature Review", *MIS Quarterly 26*(2), 2002, pp. xiii–xxiii.

8. Appendix

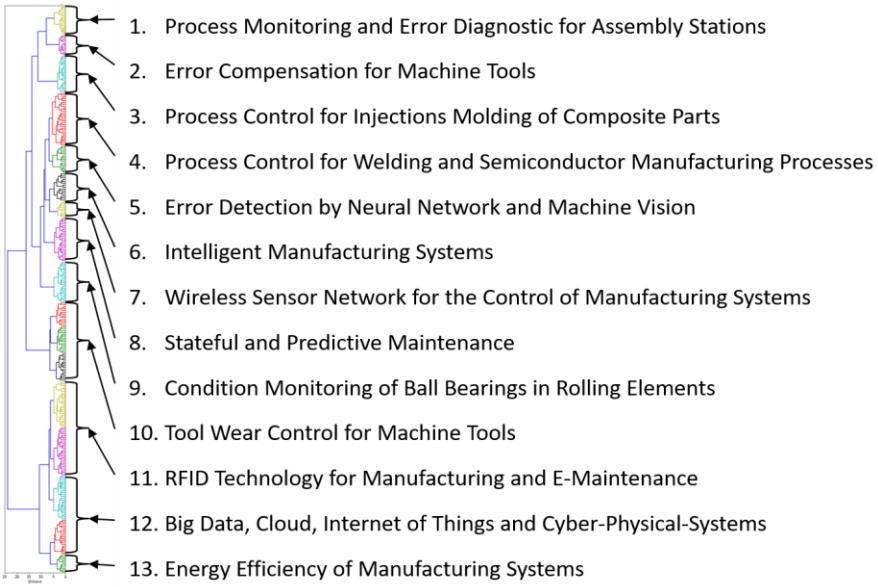

Figure 7. Dendrogram of all publications with cluster titles

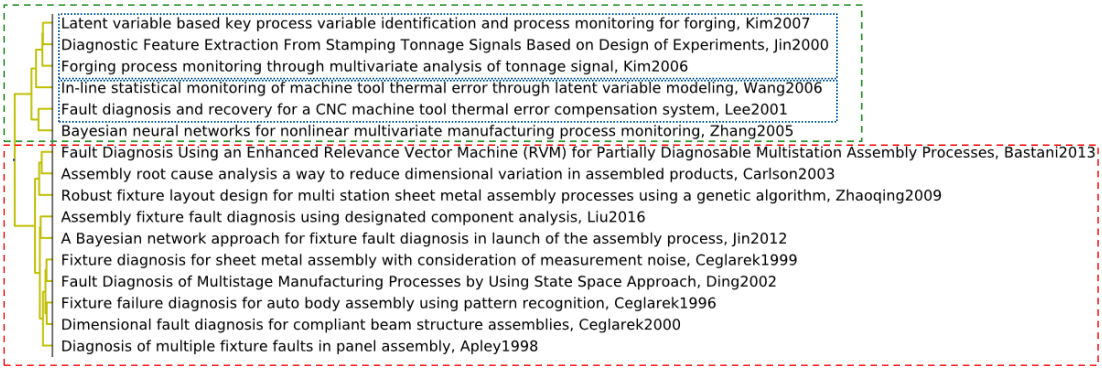

Figure 8. Dendrogram with titles and authors for cluster 1

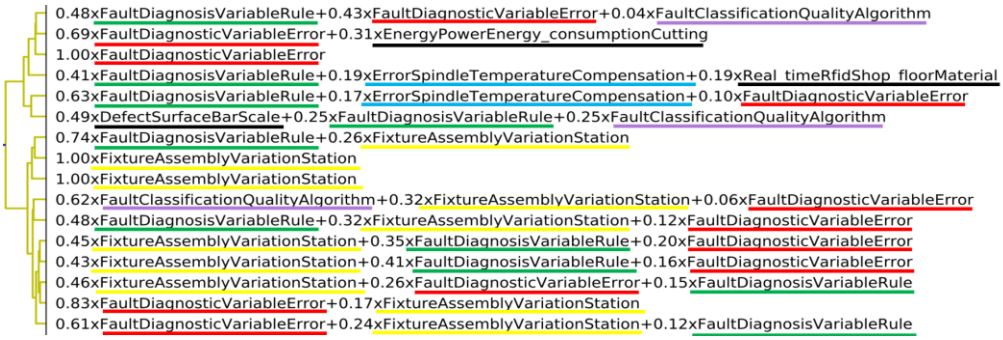

Figure 9. Dendrogram with topics of cluster 1